REMARKS

Claims 1-28 remain in the application. Claims 1, 7, 12, 19, and 24 have been amended.

Claim Rejections under 35 U.S.C. § 112, Second Paragraph

Claims 1-28 were rejected under 35 U.S.C. § 112, second paragraph as failing to distinctly claim the invention. These claims have been amended to recite that the storage of instructions in the instruction cache is based on the execution unit or cluster that the instruction requires. In view of this amendment, reconsideration and withdrawal of the rejection of claims 1-28 under 35 U.S.C. § 112, second paragraph is respectfully requested.

Claim Rejections under 35 U.S.C. § 102

Claims 1-12, 14-24 and 26-28 were rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,790,822 to Sheaffer et al. ("Sheaffer"). Claims 1, 2, 7, 8, 12, 19, 20, and 24 were rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,471,593 to Branigin ("Branigin"). Claims 12-18 and 24-28 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,219,780 to Lipasti ("Lipasti").

As amended in response to the § 112, second paragraph rejection, the independent claims recite the steering of instructions into an instruction cache so as to store them in the cache based on the execution unit or cluster that the instruction requires. Looking at Fig. 2, for example, instructions are steered and stored into bins of the instruction cache where each bin is assigned to a particular execution unit. As an example, an instruction requiring execution by a floating point execution unit would be steered and stored in the instruction cache (e.g., into a particular bin) based on the floating point execution unit that is required for the instruction's execution.

Sheaffer names the same inventor and assignee as the present application. Looking at Col. 6, lines 6-18 and Figs. 1 and 4, a re-order unit 134 is provided that places instructions into a different order before being stored in the instruction cache. The cited text is reproduced below:

> In FIG. 4, an overview of one embodiment of a re-order unit 134 is illustrated. The re-order unit 134 includes an input buffer 410 which receives in-order instructions from memory 114. A first selection logic 415 channels the instructions from the input buffer 410 to separate resource bins 422-426 based on the execution resources needed by the individual instructions. Once the instructions are channeled into the resource bins 422-426, a second selection logic 428 takes the instructions out of the bins based on data dependencies. During each clock cycle an instruction is taken from each bin in accordance with the second selection logic and sent to shift register 430 for assembling cache lines in the re-ordered instruction cache 136.

As seen from the text, the second selection logic places the instructions into a shift register 430 based on data dependencies before being stored in the instruction cache 136. Accordingly, the instructions are not stored in the instruction cache based on a particular execution unit needed for that instruction as recited in the claims. Instead, the instructions are stored in a convenient order so that instructions that can be immediately executed can be executed rather than wait to execute an instruction because needed data operands are not yet valid. Furthermore, the steering logic described in the background section is still needed to steer the instructions from the instruction cache to the execution units.

Branigin describes in Fig. 5 a processor with Tomasulo's algorithm. As shown in Fig. 5, instructions are decoded and then immediately transmitted to reservation stations. The full description of the algorithm is found at U.S. 3,462,744 (Note the endnote at Col. 6, line 63 referring to *Toma1* at the top of Col. 129). Referring to the '744 patent, column 26, lines 17 et seq., instructions stored in the I box 134 are decoded to figure out what execution unit is needed (e.g., floating point execution unit). If the unit is available and the data operands are available,

the instruction is sent to the reservation station of the execution unit. Otherwise, the instruction remains at the I box until those conditions are met (Col. 26, lines 41-43). A reservation station is a component quite different from an instruction cache. A reservation station is a buffer that stores only instructions for their dedicated execution unit along with the data operands. An instruction cache, on the other hand is an associative array that allows for comparison (e.g., of tags) to determine if an instruction is present or not in the cache. Accordingly, it is improper to interpret reservations stations the same as an instruction cache.

Finally, Lipasti describes a circuit for dispatching instructions to clusters of execution units. Claims 12 and 24 each recite that instructions are steered into the instruction cache and stored based on the execution unit cluster that the instruction requires. Clearly Lipasti discloses an instruction cache as element 40, and all steering decisions occur after the instructions are stored in the instruction cache. For much of the same reasons as presented above with respect to reservation units, the "windows" (e.g., element 62) are not the same as an instruction cache. Their functionality is quite different, and Lipasti clearly teaches that the element 62 is a part, wholly different than the instruction cache called out in Figs. 3 and 4, for example (it is noted that at Col. 7, lines 1-3, Lipasti includes "reservation station," but doesn't include a cache as a type of storage component for the windows, even though the instruction cache is mentioned in the same paragraph).

In view of the above, reconsideration and withdrawal of the rejection of claims 1-28 under 35 U.S.C. §§ 102(b) and 102(e) is respectfully requested.

CONCLUSION

For all the above reasons, the Applicant respectfully submits that this application is in condition for allowance. A Notice of Allowance is earnestly solicited.

The Examiner is invited to contact the undersigned at (202) 220-4255 to discuss any matter concerning this application. The Office is hereby authorized to charge any additional fees or credit any overpayments under 37 C.F.R. § 1.16 or § 1.17 to Deposit Account No. 11-0600.

Respectfully submitted,
KENYON & KENYON

Dated: 10/30/03

By: _____
Shawn W. O'Dowd
Reg. No. 34,687

KENYON & KENYON
1500 K Street, NW
Suite 700
Washington DC, 20005
(202) 220-4200 telephone
(202) 220-4201 facsimile
DC:473729v.1